Joint Program Executive Office
Joint Tactical Radio System

SCA IDL Refactorization




24-25 August 2010
JTRS SCA Working Group

JPEO JTRS

Task Overview

- **Objective:**
  - The CF.idl can be decomposed into a set of IDL files so that an SCA component uses only the interfaces which are needed for that component. The focus of this change proposal is to decompose only the IDL in the SCA v2.2.2 main document into logical groupings that are aligned by the type of components being developed

- **Benefits**
  - Smaller memory footprint for executables
  - Reduced power requirements

- **Impact**
  - Minimum Impact to existing SCA products
  - Backward Compatibility

Background / Status

- **Started with the OMG SW Radio Spec's IDL partitioning and aligned it with SCA v2.2.2 IDL**
- **Initially reviewed in July, 2009**
- **Final approval in January, 2010**
- **Next Steps**
  - Incorporate Deployment Optimization and Lightweight (LW) Components changes into final IDL set

Highlights of SCA IDL Refactorization

- **Decomposed SCA v2.2.2 IDL (CF.idl, PortTypes.idl, & StandardEvent.idl) as follows:**
  - Applications
  - CommonTypes
  - Device
  - DeviceManager
  - DomainManager
  - ExecutableDevice
  - LoadableDevice
  - ResourceFactory
  - Resources
  - File
  - FileManager
  - FileSystem
  - Unique Port Types for each Sequence Type
  - DomainEvents
  - StateEvents

Backward Compatibility

- **Provided "Condensed IDL" files to maintain backward compatibility**
  - CF.idl example below …

```
//Source file: CF.idl

#ifndef __CF_DEFINED
#define __CF_DEFINED

/* This file is provided to maintain backward compatibility with
   legacy systems that use CF.idl files */

#include "CFCommonTypes.idl"
#include "CFResources.idl"
#include "CFResourceFactory.idl"
#include "CFDevice.idl"
#include "CFLoadableDevice.idl"
#include "CFExecutableDevice.idl"
#include "CFApplications.idl"
#include "CFDomainManager.idl"

#endif
```

Executable Size Comparison

Test Case	Executable Size (bytes)	Delta Size from Server Mainline (bytes)	% Change (from Original IDL)
1. Server Mainline Only	1,146,272	N/A	N/A
2. Server + Resource Object Original SCA IDL	1,591,148	444,876	N/A
3. Server + Resource Object Condensed IDL	1,618,452	472,180	+6%
4. Server + Resource Object Resource Only IDL	1,301,656	155,384	-65%

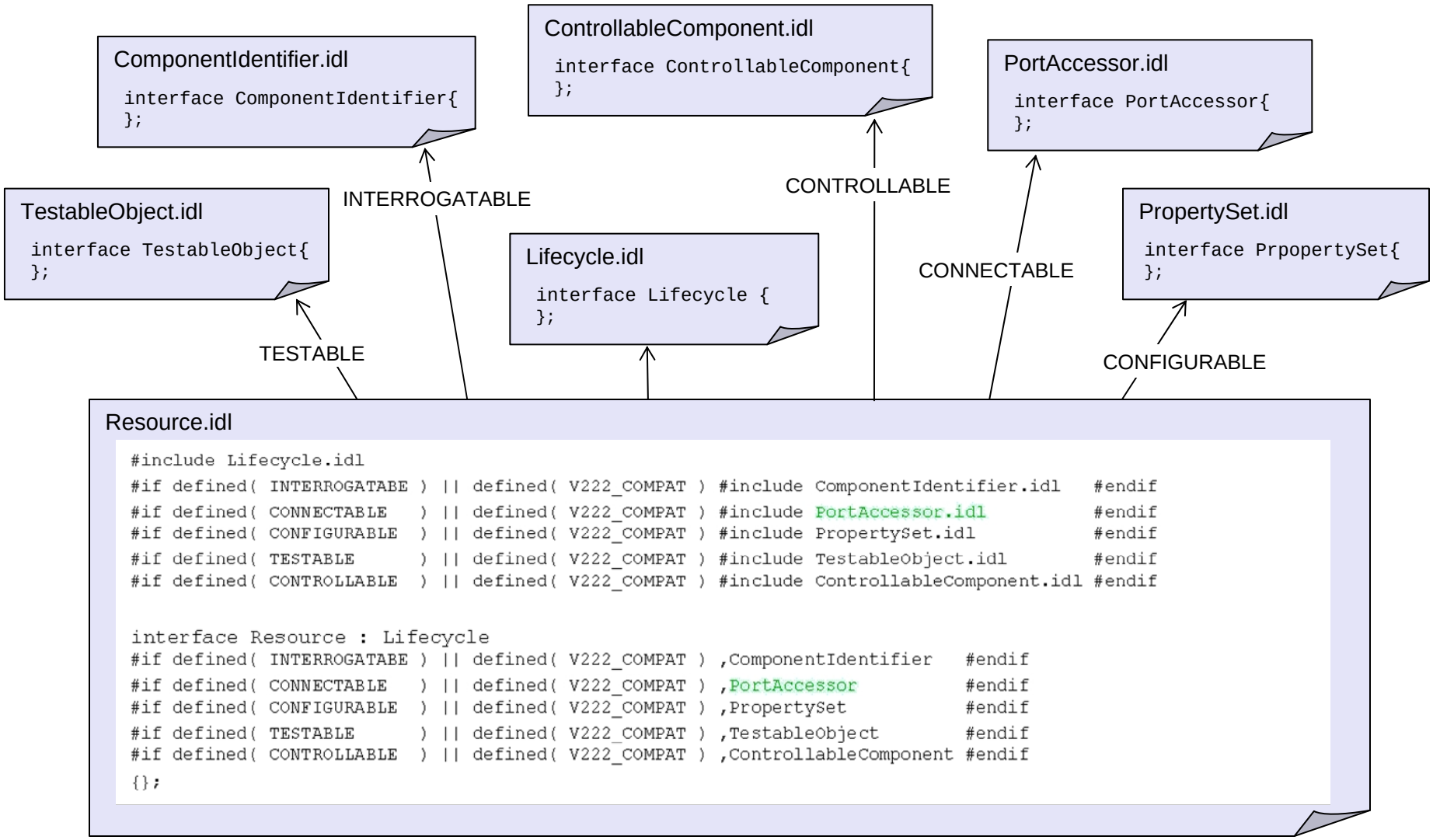
IDL Changes for Deployment Optimizations & Lightweight Components Tasks

Deployment Optimizations & Lightweight Components Tasks - IDL Changes

- Applications
- CommonTypes
- Device
- DeviceManager
- DomainManager
- ExecutableDevice
- LoadableDevice
- ResourceFactory
- Resource
- ComponentIdentifier
- ControllableComponent
- Lifecycle
- PropertySet
- PortAccessor
- TestableObject
- File
- FileManager
- FileSystem
- Unique Port Types for each Sequence Type
- DomainEvents
- StateEvents

Interfaces impacted

IDL Files


ComponentIdentifier.idl
interface ComponentIdentifier{
};
ControllableComponent.idl
interface ControllableComponent{
};
PortAccessor.idl
interface PortAccessor{
};
TestableObject.idl
interface TestableObject{
};
Lifecycle.idl
interface Lifecycle {
};
PropertySet.idl
interface PrpopertySet{
};
INTERROGATABLE
CONTROLLABLE
CONNECTABLE
TESTABLE
CONFIGURABLE
Resource.idl
#include Lifecycle.idl
#if defined(INTERROGATABE) || defined(V222_COMPAT) #include ComponentIdentifier.idl #endif
#if defined(CONNECTABLE) || defined(V222_COMPAT) #include PortAccessor.idl #endif
#if defined(CONFIGURABLE) || defined(V222_COMPAT) #include PropertySet.idl #endif
#if defined(TESTABLE) || defined(V222_COMPAT) #include TestableObject.idl #endif
#if defined(CONTROLLABLE) || defined(V222_COMPAT) #include ControllableComponent.idl #endif
interface Resource : Lifecycle
#if defined(INTERROGATABE) || defined(V222_COMPAT) ,ComponentIdentifier #endif
#if defined(CONNECTABLE) || defined(V222_COMPAT) ,PortAccessor #endif
#if defined(CONFIGURABLE) || defined(V222_COMPAT) ,PropertySet #endif
#if defined(TESTABLE) || defined(V222_COMPAT) ,TestableObject #endif
#if defined(CONTROLLABLE) || defined(V222_COMPAT) ,ControllableComponent #endif
{};

BACK UP CHARTS

Test Cases

1. **Server Mainline Only**
   - i.e. orb_init

2. **Skeleton Implementation of Resource Object activated on Root POA using original CF IDL**

3. **Skeleton Implementation of Resource Object activated on Root POA using condensed CF IDL**

4. **Skeleton Implementation of Resource Object activated on Root POA using CF Resource IDL only**